Claims

1. A method of modular multiplication of a multiplicand by a multiplier, in which a modulus is employed, making use of a multiplication look-ahead process and a reduction look-ahead process, said method comprising the steps of:

transforming the modulus into a transformed modulus that is greater than the modulus, with a predetermined fraction of the transformed modulus having a higher-order digit with a first predetermined value that is followed by at least one lower-order digit having a second predetermined value;

iterative working off of the modular multiplication making use of the multiplication look-ahead process and the reduction look-ahead process and utilizing the transformed modulus so as to obtain at the end of the iteration a transformed result for the modular multiplication; and

re-transforming the transformed result by modular reduction of the transformed result utilizing the modulus.

2. A method according to claim 1,wherein the step of iterative working off comprises a plurality of iteration steps, with a multiplication intermediate result and a reduction shift value being determined in one of said iteration steps, with the reduction shift value being computed using a determination of the number of digits between the higher-order digit with the first predetermined value of the transformed

modulus and the highest-order digit of the intermediate result having said first predetermined value.

3. A method according to claim 2, wherein a multiplication shift value is determined in said multiplication look-ahead process, and wherein a reduction shift value for the reduction look-ahead process is calculated by subtraction of said predetermined number of digits from the multiplication shift value.

4. A method according to any of the preceding claims, wherein said step of iterative working off comprises the following steps:

in a first iteration step:

(a) performing a multiplication look-ahead process to obtain a multiplication shift value;

(b) multiplying a base raised to the power of the multiplication shift value by a current intermediate result to obtain a shifted intermediate result;

(c) performing a reduction look-ahead process to obtain a reduction shift value by determining an auxiliary shift value equal to the number of digits between the higher-order digit with the first predetermined value of the predetermined fraction of the transformed modulus and the highest-order digit of the intermediate result having said first predetermined value, and by calculating the reduction shift value using the auxiliary shift value and the multiplication shift value;

(d) multiplying the transformed modulus by the base raised to the power of the reduction shift value to obtain a shifted transformed modulus; and

(e) summing the intermediate result and the multiplicand and subtracting the shifted transformed modulus to obtain an updated intermediate result.

5. A method according to claim 1, wherein said predetermined fraction of the modulus is 2/3.

6. A method according to claim 5, wherein the multiplicand, the multiplier and the modulus are binary, with the base being 2, and wherein the higher-order digit of the predetermined fraction of the transformed modulus has a first predetermined value of 1 and the at least one low-order digit has a second predetermined value of 0.

7. A method according to claim 6, wherein the most significant bit of the transformed modulus is a sign bit, and a higher-order section of the predetermined fraction of the modulus reads as follows:

01000 xx ... xx,

in which the bits designated xx may have arbitrary values.

8. A method according to claim 7, wherein the higher-order section of the transformed modulus reads as follows:

01100 ... 00.

9. A method according to claim 1, wherein said step of transforming the modulus comprises randomization of the modulus so that the transformed modulus is randomized.

10. A processor for modular multiplication of a multiplicand by a multiplier, in which a modulus is employed, making use of a multiplication look-ahead process and a reduction look-ahead process, comprising:

a means for transforming the modulus into a transformed modulus that is greater than the modulus, with a predetermined fraction of the transformed modulus having a higher-order digit with a first predetermined value that is followed by at least one lower-order digit having a second predetermined value;

a means for iterative working off the modular multiplication making use of the multiplication look-ahead process and the reduction look-ahead process and utilizing the transformed modulus so as to obtain at the end of the iteration a transformed result for the modular multiplication; and

a means for re-transforming the transformed result by modular reduction of the transformed result utilizing the modulus.

11. A processor according to claim 10,
comprising a host CPU and a coprocessor, said means for transforming the modulus being arranged in the host CPU and said means for iterative working off of the modular multiplication being arranged in the coprocessor.

12. A processor according to claim 11,
wherein the host CPU is a short-number arithmetic-logic unit having a number of digits smaller than or equal to 64, and wherein the coprocessor is a long-number arithmetic-logic unit having a number of digits greater than or equal to 512.

13. A processor according to claim 10,
wherein the means for iterative working off the modular multiplication comprises a register for the transformed modulus and a register for an intermediate result of the modular multiplication.